Europäisches Patentamt
European Patent Office
Office européen des brevets

Bescheinigung Certificate Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. Patent application No. Demande de brevet n°

02080318.5

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

R C van Dijk

Anmeldung Nr:
Application no.: 02080318.5
Demande no:

Anmeldetag:
Date of filing: 16.12.02
Date de dépôt:

Anmelder/Applicant(s)/Demandeur(s):

Koninklijke Philips Electronics N.V.
Groenewoudseweg 1
5621 BA Eindhoven
PAYS-BAS

Bezeichnung der Erfindung/Title of the invention/Titre de l'invention:
(Falls die Bezeichnung der Erfindung nicht angegeben ist, siehe Beschreibung.
If no title is shown please refer to the description.
Si aucun titre n'est indiqué se referer à la description.)

Device for modifying the time base of an MPEG signal

In Anspruch genommene Priorität(en) / Priority(ies) claimed /Priorité(s) revendiquée(s)
Staat/Tag/Aktenzeichen/State/Date/File no./Pays/Date/Numéro de dépôt:

Internationale Patentklassifikation/International Patent Classification/
Classification internationale des brevets:

H04N7/26

Am Anmeldetag benannte Vertragstaaten/Contracting states designated at date of filing/Etats contractants désignées lors du dépôt:

AT BE BG CH CY CZ DE DK EE ES FI FR GB GR IE IT LI LU MC NL PT SE SI SK

Device for modifying the time base of an MPEG signal

The invention relates to a device for modifying the time-base of an MPEG signal. The invention further relates to a method for modifying the time-base of an MPEG signal. Furthermore the invention relates to an storage device comprising a device according to the invention.

Now a days a lot of people have analog video signals recorded on tapes. There is a need to convert these analog video signals to digital video signals, for example an MPEG signal, and to record the digital video signals on a digital recording medium such as Harddisk, DVD-discs. To be able to so, the user reproduces his analog video signal with his analog reproducing device, such as a VCR and Camcorder so as to obtain the analog video signal and encodes this signal with an MPEG encoder. The MPEG encoder could be part of a television. Via a network conneotion, such as P1394 interface, the generated MPEG signal is supplied to the digital storage device. The MPEG signal supplied via the network connection should be MPEG compliant. For the PCR in the MPEG signal this means that the deviation should less than 30 ppm. Analog video signals should normally have a frame rate of 25 Hz or 30 Hz and a vsync pulse with the same frequency. Conversion of an analog video signal having a frame rate of exactly 25 HZ will result in an MPEG signal with PTS values present once every 40 ms. Normally, the analog video signal is recorded with a frame rate of 25 Hz and preferably the analog video signal is converted into a MPEG signal with PTS values present once every 40 ms so as to obtain a stored digital video signal which could reproduced by any MPEG compliant decoder to obtain a video signal having a frame rate which is substantially similar the frame rate of the analog video signal previously recorded by the analog recording device. However, the deviation of the frame rate of a video signal reproduced with analog reproducing devices could deviate up to 100 ppm. To be able to obtain a digital video signal which has a PTS once every frame period, the MPEG encoder should be locked to the vsync of the video signal. The digital video signal thus obtained will have a PCR which deviates 100 ppm and is therefore not MPEG complaint as the PCR in an MPEG video signal should have a deviation less than 30 PPM. As this signal is not MPEG complaint it should not be supplied to a network as it cannot be reproduced by MPEG

decoders connected to the network. A digital storage device and a mpeg encoder are not necessarily combined but could be in different apparatuses connected to each other by a digital network. Furthermore, a digital storage device should not have necessarily have analog video inputs. To be able to record the analog recordings on a digital storage device in such an environment the analog video signal has to be encoded into an MPEG compliant signal by the MPEG encoder. The MPEG compliant signal is transmitted via the network to the digital storage device to be stored on the recording medium. When reproducing the stored MPEG compliant signal, the frame rate of the video signal will be substantially similar to the frame rate when reproduced with the VCR or Camcorder, which could deviate up to 100 ppm.

It is an object of the invention to provide a device which enables a user to any MPEG encoding device to convert an analog video signal into a MPEG compliant signal, to send the thus obtained MPEG compliant signal via an network such as p1394 to the location of a digital recording device, to convert the MPEG compliant signal in a video signal which signal when reproduced after recording on a recording device results in a MPEG compliant signal having a frame rate which is substantially equal to a desired frame rate, which frame rate is normally the frame rate equal to the frame rate of the original analog recording.

The device according to the invention is characterized in that it comprises:

- means for determining new values of the presentation time stamps in dependence the presentation time stamps;
- means for determining new values of the presentation time stamps in dependence of the program clock reference time stamps and the presentation time stamps;
- means for combining the video signal, new values of presentation time stamps and new values of program clock reference time stamps so as to obtain a modified digital video signal.

It should be noted that retiming of an MPEG signal is known from US6,101,195. The MPEG signal is retimed to avoid jumps in the timing signals when switching from a first MPEG stream to a second MPEG stream. The timing signals are modified in relation to the local timing reference. US6,101,195 does not disclose the modifying of the clock reference timestamps in dependence of the values of the PTS timestamps in the video signal.

These and other aspects of the invention will be apparent from and elucidated with reference to the embodiments hereafter in the figure description.

0.0.1 Clocking for DVHS

Figure 1 shows the clocking of encoder and decoder according to the model of MPEG. As an example, we assume PAL video and 48 KHz sampled audio sources.

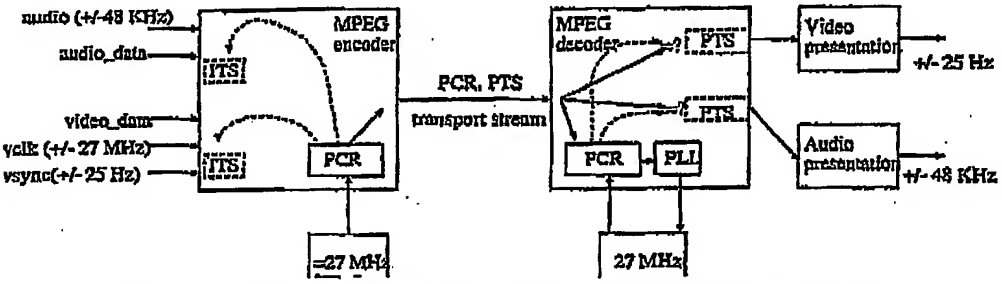


Figure 1: *Clocking strategy according to the MPEG specification*

As we can see in the figure, the encoder is clocked from an XTAL, which delivers a very precise 27 MHz signal (must be 27 MHz +–30 ppm according to MPEG). The PCR time-base counter is directly derived from this main encoder clock.

The incomming video has an associated vclk clock signal (27 MHz, line locked) and a vsync pulse (25 Hz or 30 Hz) that marks the start of a new frame. In the following we will assume disturbance in the source, in the sense that the line locked 27 MHz clock has a lot of jitter on a line by line basis (but is locked on a frame basis), and that the frequency of the vsync deviates considerably from the expected value (e.g. 25 Hz -> 24.5 Hz). Furthermore, the audio clock is assumed to deviate slightly from the desired 48 KHz. Such a situation could occur when a VCR at playback (or even worse: trickmode) is connected to the encoder. Note, that because the incomming video and the encoder itself use a different clock, there must be a clock-domain bridge at the video input of the encoder.

The audio data comes from an A/D converter, that is clocked by a free running 48 KHz signal. Because the incomming audio uses another clock than the encoder, there must be a clock-domain bridge at the audio input of the encoder.

At the input of the encoder, audio and video access units are time-stamped with ITS timestamps equal to the value of the PCR timebase at the time at which an access unit enters the chip. From these ITS time-stamps the presentation timestamps (PTS) of the access units are derived by adding a constant end-to-end delay (see Section 1.1). Because of the fact, that the video vsync is not locked to the main clock, the difference between the PTS's of two successive video access units is generally not precisely equal to the

expected 40 ms (in the example of 24.5 Hz it is smaller). Similarly, the difference between 2 audio access units will generally not be precisely equal to the expected value, because the audio clock is not locked to the PCR clock.

The multiplexer inserts a sample of the PCR timebase into the stream at regular intervals. In the decoder the received values of the PCR are used to reconstruct the encoder PCR counter. According to MPEG, the decoder can only reconstruct the PCR counter if the PCR clock is within the MPEG specifications (27 MHz +– 30 ppm). Since this is the case in Figure 1, we can assume that the reconstruction is perfect i.e. the PCR counter in the decoder is a copy of the PCR counter in the encoder.

In the decoder, the PTS's that are coded in the stream are used to determine the presentation time of access units in relation to the value of the reconstructed PCR counter. Because of the fact that the presentation times are not locked to the PCR, in practice this means that the audio and video decoders must use a different clock as the reconstructed PCR clock. The output of audio and video is a precise reconstruction of the audio and video at the input of the MPEG chain.

According to MPEG, the encoding solution of Figure 1 is acceptable. However, in practice, problems can occur if the encoder is connected to a DVHS recorder or a set-top-box (STB). This has to do with the implementation of the DVHS recorder and the implementation of practical STB's.

We will now shortly explain the characteristics of these devices.

Characteristics of the DVHS recorder

Figure 2 shows a block diagram of the DVHS recorder.
At record, the main clock is derived from the PCR values that come into the recorder via the MPEG transport stream. Basically, this procedure is the same as in the standard MPEG decoder. However, the PLL in the DVHS recorder has a much wider range than the +– 30 ppm of MPEG.

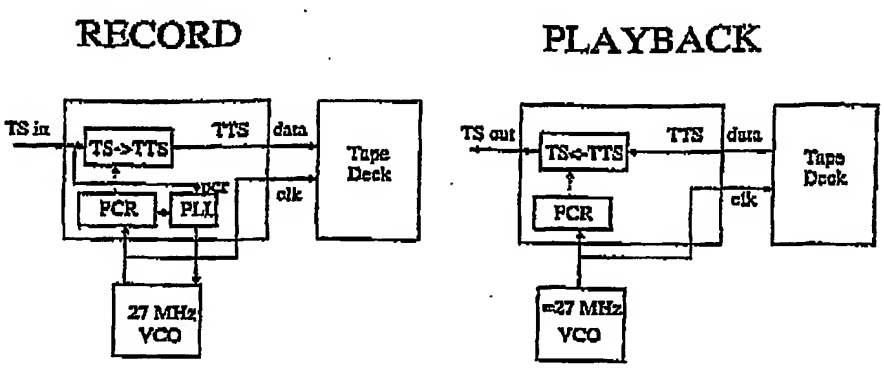


Figure 2: *Clocking in the DVHS recorder*

All the processing in the DVHS is done by using the PCR-locked main clock. In fact, also the mechanics of the tape deck are locked to the main clock. This implies that a certain number of tracks on tape have a fixed duration in time in relation to the main clock. For example, 12 tracks on tape for PAL correspond to exactly (0.2 s * main_clock_frequency) cycles of the main clock. All TS packets that come into the recorder are time-stamped from the pcr timebase in the TS->TTS block. The resulting stream, called a time-stamped transport stream (TTS), contains 4 bytes of timestamp data and 188 bytes of video data per packet.

At playback, the DVHS recorder uses a very precise 27 MHz clock, within the 30 ppm limits of MPEG, for all of the electronics and mechanics. For each TS packet, the time-stamp that was added during recording is used to determine the time it has to be sent out in the TTS->TS block. This is done by comparing the time-stamp with a counter that is clocked by the 27 MHz clock. The TTS->TS block contains a sufficient amount of buffer memory to hold data that is not yet meant to go out.

An [redacted], is that the recorded video must be editable at the GOP boundaries. Since, physically, editing can only be done at track boundaries, in the editable mode a video GOP must fit exactly into an integer number of tracks. For PAL, a GOP of 5 has been selected to fit precisely into 12 tracks. GOP boundaries must precisely match a track boundary.

Characteristics of practical STB's

As an example of how a typical STB is implemented, we will describe the Suresnes STB. This STB is planned to be sold in combination with a DVHS recorder. In fact, in the first period after introduction it will probably be the only STB that will be able to function together with a Philips DVHS recorder. The STB and the recorder communicate with each other via p1394.

The MPEG decoder in the Suresnes STB uses only a single clock, PCR locked, for audio and video decoding. In principle, the STB expects that the audio and video clock in the encoder where both locked to the PCR, and therefore there is allways a fixed number of cycles of the PCR clock between 2 successive access units. It will start decoding of audio/video access units at the expected pace, independent from the precise PTS values that are in the stream. For example, for PAL the decoder will start decoding a new frame after every 40 ms of the PCR clock.

Before decoding of a particular frame, the decoder will check whether or not its PTS is still within a distance of 1/2 frame period from the really used decoding time Tdec (Tdec follows from the assumed fixed-pace decoding). If not, the decoder makes the following correction:

IF (Tdec - PTS > 1/2 frame_period) THEN skip_access_unit;
IF (Tdec - PTS < -1/2 frame_period) THEN repeat_access_unit;

In this way, the decoder still effectively locks the presentation times to the PTS's, but artifacts will occur at regular times if the audio and video were not already locked to the PCR.

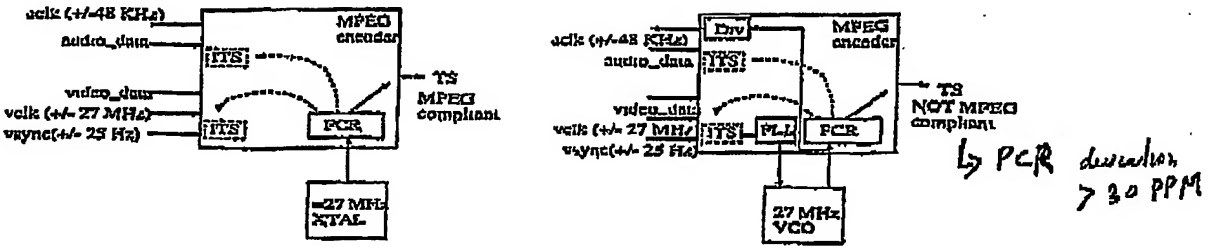
From these characteristics, it follows that the following problems occur in case the solution of Figure 1 is used to connect the MPEG encoder to a DVHS recorder.

1. **The editable format is not possible.** Since the PCR is not locked to the vsync, the difference in time between two successive GOP-starts is not equal to the expected 200ms (PAL, measured according to the PCR clock). This means that, since the mechanics of the recorder lock to the PCR, it is therefore not possible to store a GOP in an integer number of tracks.
2. **Artifacts will be visible and audible during playback on the Suresnes STB.** The incomming signal will be precisely reconstructed at playback. Since the audio and video PTS's are not locked to the PCR, the decoder will have to skip and repeat access units.

The next sections will describe methods that try to solve these problems.

0.0.1.1 Alternative clocking strategy 1: locking video, audio and PCR with a VCO

A solution to these problems is shown in Figure 3



↳ PCR deviation > 30 PPM

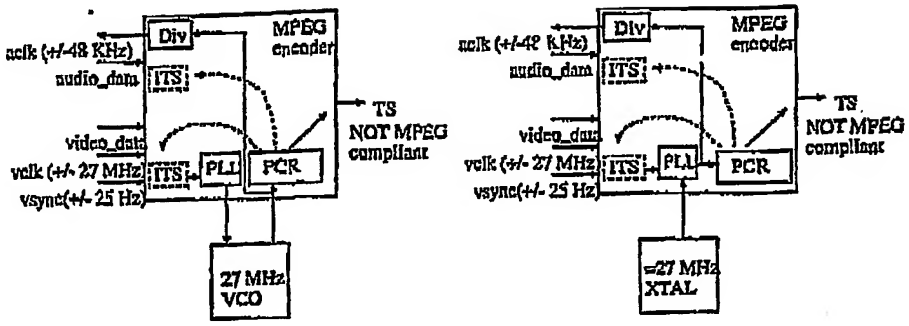
Figure 3: *Alternative solution for DVHS clocking strategy using a PLL in the encoder*

The system in this figure only differs from the MPEG model in the sense that it uses a PLL in the encoder to lock the main clock to the incomming vsyncs of the video. The PLL reads ITS timestamps from the video input, and modifies the VCO clock in a way that the difference in successive ITS's becomes equal to 40 ms. The audio encoder is clocked from a signal that is in turn derived from the PCR clock. In this way, audio and video PTS's are locked to the PCR clock. As a result of this:

1. **The editable format is possible.** The mechanics of the recorder lock to the PCR, which is again locked to the incomming video. The time between 2 GOPS, measured in cycles of the PCR clock, is constant and corresponds precisely to 200ms for PAL. Therefore each GOP exactly fits in 12 tracks.
2. **There are no artifacts at playback.** At playback the audio and video PTS's will be locked to the PCR, so no access units will be skipped/repeated. All deviations that were in the video clock before encoding will be automatically corrected at playback (time-base correction at no additional cost), because a stable 27 MHz clock is used. In the example, the video rate at playback will be precisely 25 Hz.

0.0.1.2 Alternative clocking strategy 2: locking video, audio and PCR without a VCO

In the solution of Figure , the operating range of the PLL is limited by the VCO. In Figure 4 another implementation is shown in which the locked clock is not used for the complete IC, but only for the blocks that need a locked clock : the PCR counter and the audio clock generator.



Alternative 1 ⟶ Alternative 2

Figure 4: *Clocking strategy for DVHS without VCO*

This works as follows:

The IC is now clocked with a precise 27 MHz signal derived from an XTAL. For every incomming vsync, the PLL compares the ITS with the expected ITS (which is a sample of the PCR at the time at which the access unit arrives at the input) E.g. for PAL, the expected ITS value is increased with 40ms for each frame. The difference between expected and real ITS is used to hold or speed up the incrementing of the PCR counter. If the expected value is higher than the real value, then the PCR counter is raised by 2 for some cycles. If the expected value is lower than the real value, then the PCR counter is not increased for some cycles. Basically, the PCR is now clocked by a stop-and-go clock which can have a lot of jitter on a cycle by cycle basis. The increment signal that goes to the PCR counter is also used to derive the audio clock.

In this way, the PCR counter and the audio clock are locked to the incomming vsyncs, although no VCO is needed. The range of the 'PLL' that is constructed in this way is virtually unlimited, because the PCR clock can be made very fast and very slow. However, the practical range of the PLL will be limited by the minimum amount of cycles that are necessary to compress a picture. In fact, if the vsyncs come too fast, then frames will be skipped in the front end. In practice, this will mean that the range of the PLL is limited to about +2% in the higher direction, and vitually unlimited in the lower direction.

0.0.1.3 Alternative clocking strategy 3: locking the PCR to audio and video in the recorder

A potential drawback of the last 2 solutions was the fact that the PCR, as transmitted on the p1394 channel between encoder and recorder, does not comply to the MPEG standard (more than +-30ppm is possible). This means that a decoder could have problems with decoding the stream in case it is also connected to the p1394 channel for monitoring

while recording.

The solution that is proposed here solves also this potential problem at the cost of some additional processing in the recorder for deriving an error signal for the PLL and PTS/DTS parsing.

Conceptually, the procedure can be explained best by assuming that the DVHS recorder now contains an MPEG decoder and an additional MPEG encoder. Figure 5 shows this:

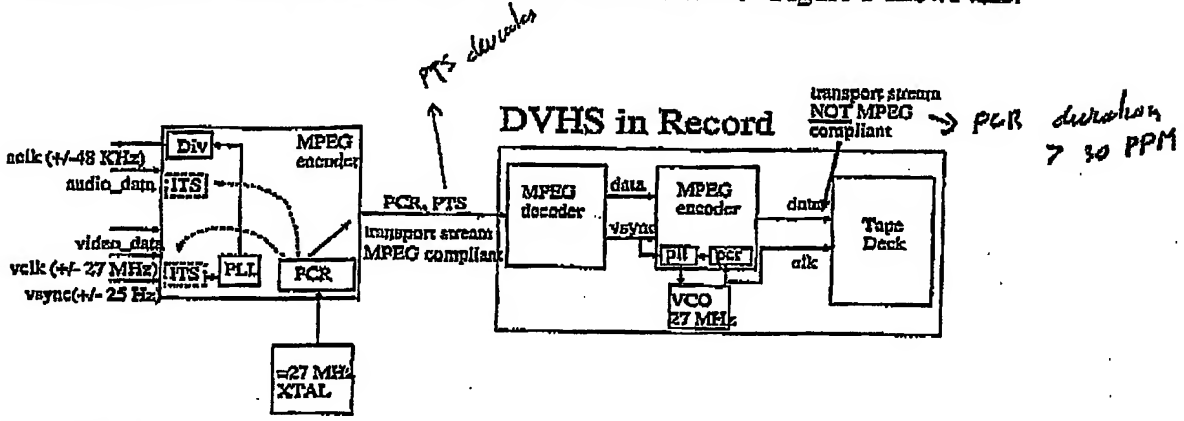


Figure 5: *Conceptual diagram of the clocking strategy acording to alternative 3*

At the encoder side, the audio clock is derived from the video-locked clock, exactly like in the solution of Figure 4 by using a PLL that locks onto the video ITS timestamps. However, in contrast to the previous solution, the PCR is generated from the precise 27 MHz clock and not from the video-locked clock. This means that the stream on the p1394 channel is MPEG compliant. Note, that the audio and video PTS's are again derived by adding a constant delay to the ITS's. The ITS's are samples of the PCR time-base, so they are also derived from the precise 27 MHz clock.

The multiplexing rate in the encoder should be exactly the same as for alternative 2, although the timebase is now derived from the precise 27 MHz clock. This means that the SW has to use the results from the PLL to derive the pace at which packets are sent out, so that the mux rate is locked to the video vsyncs (just like in alternative 2)

At the recoder side, the received MPEG signal is decoded by a (conceptual) MPEG decoder. Because of the MPEG compliant input stream, the decoder is able to exactly reconstruct the signal as it came into the encoder. This means that also the vsyncs that are produced by the decoder are a copy of the vsync as it came into the encoder.

Inside the recorder, there now is a second (conceptual) encoder, which reencodes the signal, thereby locking its timebase to the incomming vsyncs. The stream that comes out of this encoder is exactly the same as the stream that is produced in the encoder of alterna-

tive 2. This implies that the stored data on tape is exactly the same as for alternative 2, therefore that the data is editable, and we automatically have a time-base corrected signal at playback.

In the proposed implementation for this clocking stragegy, we don't need to put an encoder and decoder into the recorder, but we achieve the same result by only doing some minor additional SW processing on incomming PTS and PCR timestamps.

Figure 6 shows the procedure

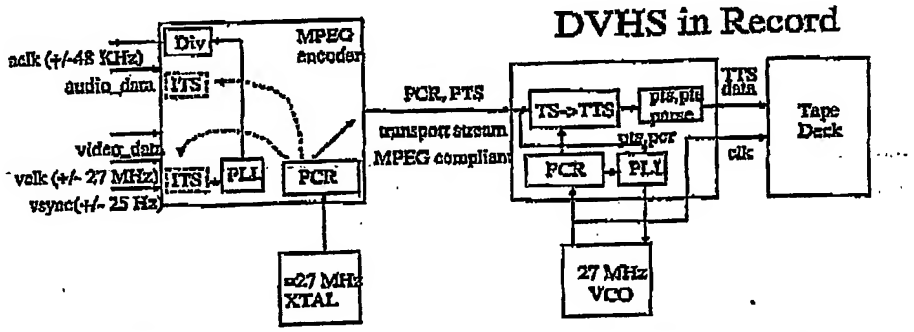


Figure 6: *Clocking strategy for DVHS with MPEG compliant data at the channel*

What changes in the recorder (as compared to the recorder in Figure 2), is that the PLL not only uses PCR samples from the TS but also PTS timestamps. Because the PTS's are derived from the ITS's, they reflect the way that video frames came into the encoder, and hence they can be used to lock the main DVHS 27 MHz clock to the vsyncs of the video as they came into the encoder.

In order to show that such an approach can be really used in practice, we will now give a possible implementation. Note, that this implementation has by no means been optimized for performance or processing load.

0.0.1.3.1 Possible implementation.

For each received PTS, we calculate an expected PTS with which the received value is compared. The first expected PTS is equal to the first received PTS. All following expected_PTS's are calculated by adding a constant frametime to the previous expected_PTS:

$$expPTS[0] = recPTS. \quad (1)$$

$$expPTS[i+1] = expPTS[i] + frametime, \quad (2)$$

where expPTS denotes an expected value of the PTS and recPTS denotes the actually

received PTS. E.g. for PAL video, a new expPTS is allways set 40ms higher than its predecessor.

For each received PCR value in the transport stream, the error signal E for the PLL is derived as:

$$E[j] = \text{timebase}[j] - T[j], \quad (3)$$

where timebase[j] is the sample of the timebase counter that was taken when receiving PCR packet j.

The target T[j] is derived as

$$T[j] = (PCR[j] - PCR[0]) \cdot \left(\frac{expPTS[k] - expPTS[0]}{recPTS[k] - recPTS[0]}\right) + PCR[0], \quad (4)$$

where j denotes the index of the received PCR and k corresponds to the number of PTS values received before the PCR value. The value of k must be bigger than 0 before the algorithm can start working. All the calculations, including the division in this equation do not have to be very precise, since the result is only used to steer the PLL. Note , that Equation 4 in essence aims at locking the PLL to the average frametime (running average from frame 0 to frame k). A more practical implementation would use a low-pass filtered version of the frametime to lock the PLL.

The error signal E is used by the PLL to modify the frequency of main 27 MHz clock. If E>0 then the frequency is decreased, if E<0 then the frequency is increased. The goal of the PLL is to control the error signal to zero.

In the recorder, the PCR[j] timestamp in the TS must be overwritten by timebase[j]. Furthermore, each received recPTS[k] must be overwritten by expPTS[0].

That above mentioned approach indeed leeds to a lock onto the vsyn can be understood by using an example that shows the start-up behaviour of the PLL. In Figure 7 the timing of a received transport stream is depicted which contains video frames that originally came into the encoder at 21.7 Hz (=1/0.046).

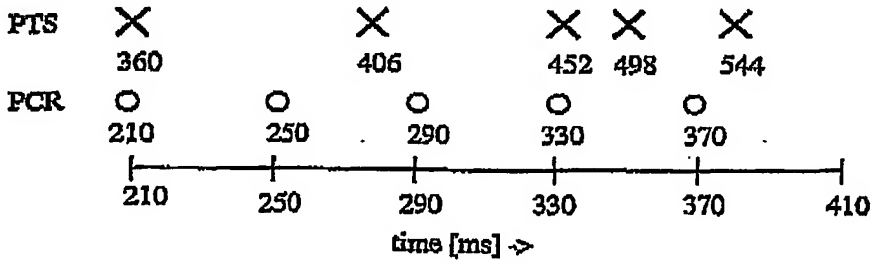


Figure 7: *Timeline for original TS containing 21.7 Hz video.*

The x-axis shows the real time, which for the example is assumed to corresponds to the time according to the precise 27 MHz timebase in the encoder. In the TS, PCR values are present once every 40 ms. The first PCR value equals 210 ms, and since the encoder is assumed to use a perfect timebase, the values in the PCR packets are the same as the time at which they are received. 5 PTS values are present in the stream per GOP. However, at what time these PTS values are received depends on the sizes of the particular frames in the GOP. E.g. since the first frame (=Iframe) is normally the largest, the difference in receival time between the first and second PTS in the stream is normally relativley large. The values of the PTS timestamps depend on the time at which the corresponding frame was received at the input of the encoder. In our example, each successive PTS is 46 ms larger than its predecessor. The value of the first PTS is chosen in the encoder according to the desired vbv_delay.

In Figure 8 the timestamps that are modified according to the above mentioned algorithm are shown in boxes. Furthermore the value of the PLL error signal E[j] is shown.

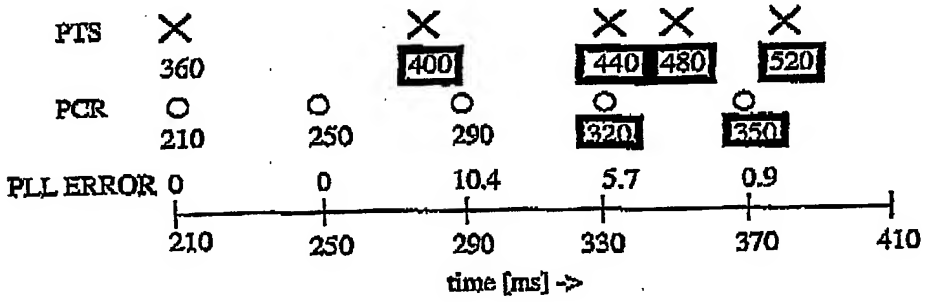


Figure 8: *Modified timestamps by using the algorithm of Equation 3.*

As we can see in the figure, all PTS's are overwritten by their expected value (40 ms increase per frame). The PLL error signal can only be calculated after 2 PTS values are received (k=1), so this is after t=280 ms. Before this the error signal is 0 and the VCO runs at an uncorrected 27 MHz. The first error signal unequal to 0 is calculated at t=290 ms. Before that time, the PCR is overwritten by a free running timebase (for now we asume again that this timebase runs exactly at 27 MHz), so that the PCR is not changed. The target T at t=290 equals:

$T = (290 - 210) * (400 - 360) / (406 - 360) + 210 = 279.6$ ms,

so the error signal equals

$E = 290 - 279.6 = 10.4$ ms

Because of this positive error signal, the PLL will slow down the clock. For the next

PCR packet, this leads to a reduced error of

$T = (330 - 210) * (440 - 360) / (452 - 360) + 210 = 314.3$ ms,
$E = 320 - 314.3 = 5.7$ ms

After some time, the PLL will have reached a stable situation with an error very close to to zero.

Figure 9 depicts the locking of the PLL.

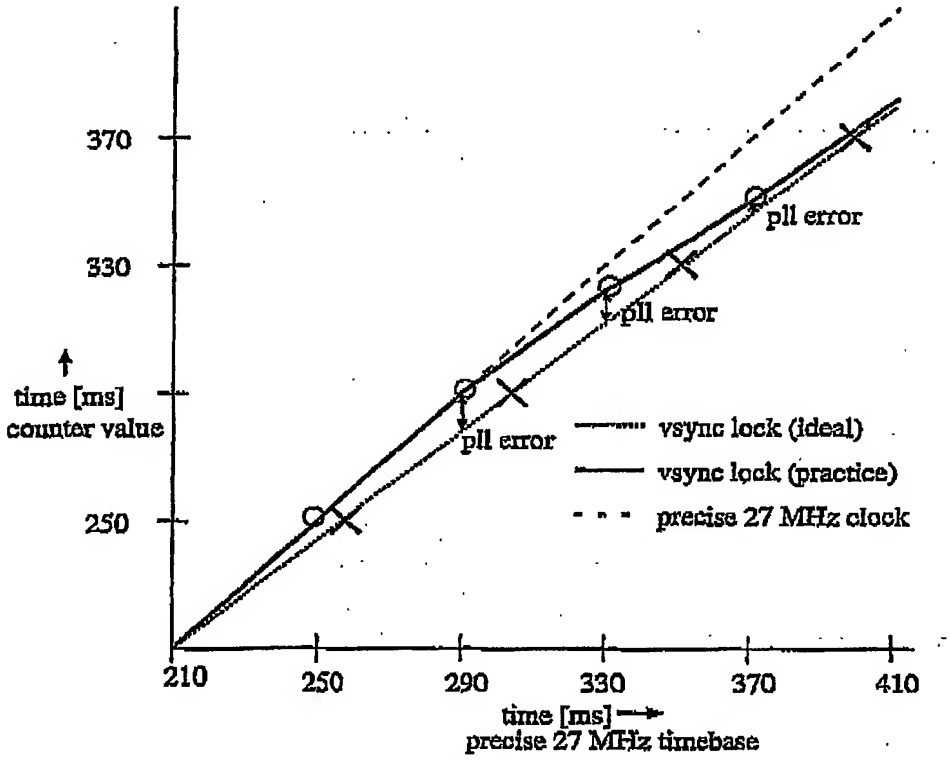


Figure 9: *Graphical representation of the locking of the PLL*

The solid line represents the value of the DVHS timebase counter as a function of the real time. Before t=290 ms, the DVHS timebase is free running at 27 MHz. After that, the PLL slows down the clock, and the DVHS timebase approaches the ideal vsync locked timebase (dotted). This ideal timebase ran at 40/46 of 27 MHz from the beginning. The crosses on the ideal curve show received PTS values normalized to the value of recPTS[0].

Once the PLL is locked, we have a situation where, at playback, we there is a timebase corrected output signal again with vsync precisely equal to 25 Hz.

Required changes in the encoder

- the audio clock must be locked to the video syncs, but the PCR must be derived from the precise XTAL clock.
- The SW must lock the MUX rate to frequency of the the video syncs. This, although the private timestamps have to be provided in relation to the precise 27MHz timebase.

Required changes in the DVHS recorder

We will describe the differences in the implementation of the recorder between alternative 1 (which is the same as alternative 2 for the recorder) and alternative 3.

Alternative 1:
HW : if PCR packet is received sample timebase counter -> timebase_sample
SW : read the PCR_value from the PCR packet and read the corresponding timebase_sample from a register
SW : calculate error signal : Error := PCR_value - timebase_sample
SW : control VCO based on error signal

Alternative 3: (Operations marked with * are modified)
HW : if PCR packet is received sample timebase counter -> timebase_sample
SW : read the PCR_value from the PCR packet and read the corresponding timebase_sample from a register
*SW : set PID filter so that packets containing PTS are filtered out (TS header contains info on presence of PTS)
*SW : if PTS packet is received, read PTS
*SW : calculate error signal according to Equation 3
SW : control VCO based on error signal
*SW : if PCR packet received -> replace PCR value with timebase_sample
if PTS packet received -> replace read_PTS with expected_PTS

If wanted, PTS's and PCR's can be put at fixed positions inside a packet by the encoder. Also the frequency of occurence for PTS's and PCR's can be set in the encoder according to implementation constraints in the recorder.

CLAIMS:

1. Device for modifying the time-base of a digital video signal, said digital video signal comprising a video signal, presentation time stamps (PTS) and program clock reference time stamps (PCR), said device comprising:
- means for receiving said digital video signal;
- means for retrieving said presentation time stamps and program clock reference time stamps from said video signal;

characterized in that said device further comprises
- means for determining new values of the presentation time stamps in dependence the presentation time stamps;
- means for determining new values of the presentation time stamps in dependence of the program clock reference time stamps and the presentation time stamps;
- means for combining the video signal, new values of presentation time stamps and new values of program clock reference time stamps so as to obtain a modified digital video signal.

2. Method of modifying the time-base of a digital video signal, said digital video signal comprising a video signal, presentation time stamps (PTS) and program clock reference time stamps (PCR), said method comprising the steps of:
- receiving said digital video signal;
- retrieving said presentation time stamps and program clock reference time stamps from said video signal;

characterized in that said method further comprises the steps of:
- determining new values of the presentation time stamps in dependence the presentation time stamps;
- determining new values of the presentation time stamps in dependence of the program clock reference time stamps and the presentation time stamps;
- combining the video signal, new values of presentation time stamps and new values of program clock reference time stamps so as to obtain a modified digital video signal.

3. Apparatus for reproducing a digital video signal recorded on a record carrier comprising a device according to claim 1.